**ALGORITHM AND DATA STRUCTURE PRACTICUM**

**MODULE 5**

**SORTING**



**CREATED BY :**

**KURNIAWAN BAGASKARA**

**L200214253**

**INFORMATICS STUDY PROGRAM**

**FACULTY OF COMMUNICATION AND INFORMATION SCIENCE**

**MUHAMMADIYAH SURAKARTA UNIVERSITY**

**1.**

class MhsTIF(object):

listKuliah = []

def \_\_init\_\_(self, nama, NIM, kota, us):

self.nama = nama

self.NIM = NIM

self.kotaTinggal = kota

self.uangSaku = us

c0 = MhsTIF('Ika',153,'Sukoharjo', 240000)

c1 = MhsTIF('Budi',120,'Sragen', 230000)

c2 = MhsTIF('Ali',22,'Surakarta', 250000)

c3 = MhsTIF('Caca',180,'Surakarta', 235000)

c4 = MhsTIF('Eka',47,'Boyolali', 240000)

c5 = MhsTIF('Kamidi',131,'Salatiga', 250000)

c6 = MhsTIF('Deni',132,'Klaten', 245000)

c7 = MhsTIF('Ngatiyem',50,'Wonogiri', 245000)

c8 = MhsTIF('Sumanto',23,'Klaten', 245000)

c9 = MhsTIF('Hamid',64,'Karanganyar', 270000)

c10 = MhsTIF('Yetno',70,'Purwodadi', 265000)

Daftar = [c0,c1,c2,c3,c4,c5,c6,c7,c8,c9,c10]

#Nomor 1

def swap(A, p, q):

tmp = A[p]

A[p] = A[q]

A[q] = tmp

def sortNIM(daftar):

n = len(daftar)

for i in range(n-1):

for j in range(n-i-1):

if daftar[j].NIM > daftar[j+1].NIM:

swap(daftar, j, j+1)

def checkNIM(a):

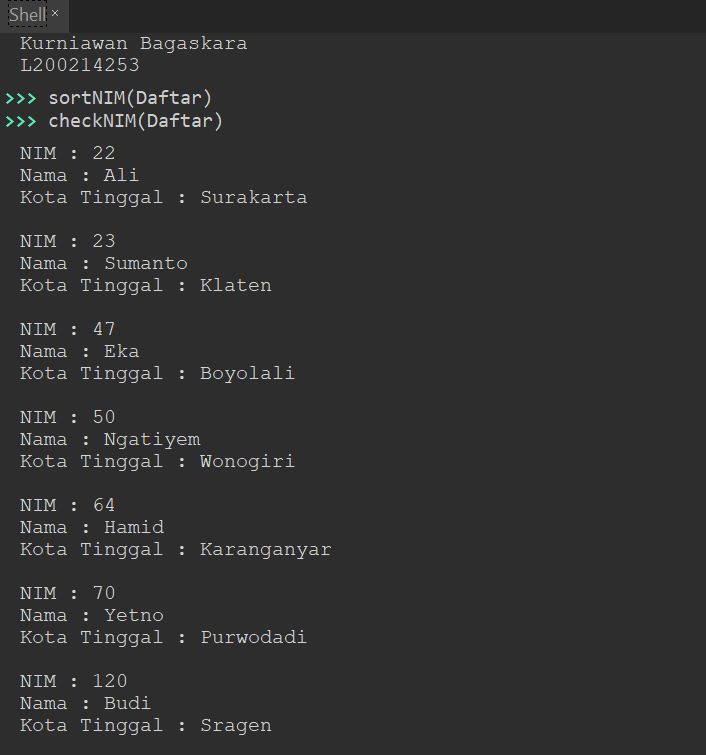
n = len(a)

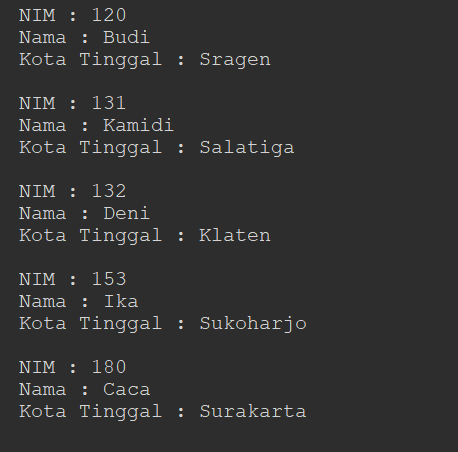
for i in a :

print('NIM : {} \nNama : {} \nKota Tinggal : {}\n'.format(i.NIM, i.nama, i.kotaTinggal))

print ('Kurniawan Bagaskara')

print ('L200214253')





**2.**

A = [2,9,78,65,5,70]

B = [1,4,12,43,22,11,120]

def sortToC(a, b):

c = a+b

for i in range(1, len(c)):

nilai = c[i]

pos = i

while pos > 0 and nilai < c[pos - 1]:

c[pos] = c[pos-1]

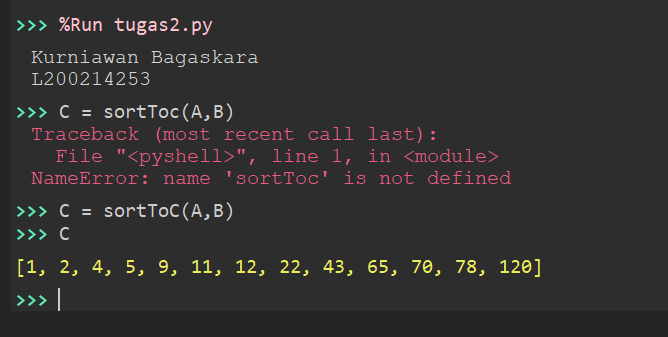
pos -=1

c[pos] = nilai

return c

print ('Kurniawan Bagaskara')

print ('L200214253')



3.

from time import time as detak

from random import shuffle as acak

def swap(A, p , q):

tmp = A[p]

A[p] = A[q]

A[q] = tmp

def cariPosisiYangTerkecil(A, dariSini, sampaiSini):

posisiYangTerkecil = dariSini

for i in range(dariSini+1, sampaiSini):

if A[i] < A[posisiYangTerkecil]:

posisiYangTerkecil = i

return posisiYangTerkecil

def bubbleSort(A):

n = len(A)

for i in range(n - 1):

for j in range(n - i - 1):

if A[j] > A[j+1]:

swap(A, j,j+1)

def selectionSort(A):

n = len(A)

for i in range(n - 1):

indexKecil = cariPosisiYangTerkecil(A, i, n)

if indexKecil != i :

swap(A, i, indexKecil)

def insertionSort(A):

n = len(A)

for i in range(1, n):

nilai = A[i]

pos = i

while pos > 0 and nilai < A[pos - 1]:

A[pos] = A[pos - 1]

pos = pos - 1

A[pos] = nilai

x = [i for i in range(1, 6001)]

acak(x)

u\_bub = x[:]

u\_sel = x[:]

u\_ins = x[:]

aw = detak();bubbleSort(u\_bub);ak=detak();print('bubble: %g detik' %(ak-aw));

aw = detak();selectionSort(u\_sel);ak=detak();print('selection: %g detik' %(ak-aw));

aw = detak();insertionSort(u\_ins);ak=detak();print('insertion: %g detik' %(ak-aw));

print ('Kurniawan Bagaskara')

print ('L200214253')

